Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	1256	711/154.ccls.	USPAT	OR	ON	2005/09/05 17:12
L2	455	711/158.ccls.	USPAT	OR	ON	2005/09/05 17:12
L3	406	370/394.ccls.	USPAT	OR	ON	2005/09/05 17:12
L4	63875	"712"/("218,225").ccls.	USPAT	OR	ON	2005/09/05 17:13
L5	55159	"709"/("216,229").ccls.	USPAT	OR	ON	2005/09/05 17:13
L6	107625	1 or 2 or 3 or 4 or 5	USPAT	OR	ON	2005/09/05 17:13
L7	11329	packet adj processor? or process$3 adj3 packet?	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/09/05 17:24
L8	11956118	@ad<"20011023"	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/09/05 17:14
L9	8822	write with before with read	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/09/05 17:15
L10	14057	(sequence adj number?) or (timestamp?)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/09/05 17:15
L11	41189	(sequence adj number or sequence adj numbers) or (timestamp or timestamps or time-stamp or time-stamps)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/09/05 17:16
L12	44	restart$3 with sequence adj number?	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/09/05 17:17
L13	240	restart$3 with L11	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/09/05 17:18
L14	22	7 and 13	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/09/05 17:18
L15	22	7 and 14	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/09/05 17:18
L16	3	6 and 15	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/09/05 17:18
L17	3	8 and 16	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/09/05 17:19

L18	1085	restart$3 near3 read	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/09/05 17:20
L19	128	6 and 8 and 18	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/09/05 17:21
L20	11	18 same 11	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/09/05 17:29
L21	6	8 and 20	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/09/05 17:23
L22	128	6 and 19 and 8	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/09/05 17:23
L23	31284	18 an d22	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/09/05 17:24
L24	128	18 and 22	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/09/05 17:24
L25	0	12 and 24	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/09/05 17:24
L26	22039	packet adj processor? or process$3 adj3 packet? or network adj process$3	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/09/05 17:24
L27	29847	24 an d26	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/09/05 17:24
L28	4	24 and 26	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/09/05 17:24
L29	4	18 with 11	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/09/05 17:30
L30	0	13 and 19	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/09/05 17:31
L31	669	711/154.ccls.	US-PGPUB	OR	ON	2005/09/05 17:31
L32	147	8 and 31	US-PGPUB	OR	ON	2005/09/05 17:31
L33	0	11 and 18 and 32	US-PGPUB	OR	ON	2005/09/05 17:31

L34	5	11 and 32	US-PGPUB	OR	ON	2005/09/05 17:31

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	52672	"711"/("154,158,221").ccls.	USPAT	OR	ON	2005/09/05 17:45
L2	11956118	@ad<"20011023"	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/09/05 17:45
L3	1085	restart$3 near3 read	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	ON	2005/09/05 17:46
L4	41189	(sequence adj number or sequence adj numbers) or (timestamp or timestamps or time-stamp or time-stamps)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/09/05 17:46
L5	11	3 SAME 4	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/09/05 17:46
L6	0	1 AND 5	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/09/05 17:46
L7	73	1 and 2 and 3	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/09/05 17:47
L8	3476	memory near5 conflict$3	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/09/05 17:47
L10	0	3 same 8	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/09/05 17:48
L11	13	3 and 8	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/09/05 17:49
L12	4	1 and 11 and 2	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/09/05 17:48
L13	0	4 and 11	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	ON	2005/09/05 17:49

Terms used **network processor dependency** or **dependencies sequence**

Found **15,074** of **148,786**

Sort results by relevance

Display results expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown

Relevance scale

1 Session 7: embedded system techniques (2): Handling of packet dependencies: a critical issue for highly parallel network processors

Stephen Melvin, Yale Patt

October 2002 **Proceedings of the 2002 international conference on Compilers, architecture, and synthesis for embedded systems**

Full text available: pdf(221.66 KB) Additional Information: full citation, abstract, references, citings, index terms

Network processors are being asked to perform increasingly complex operations on packets of information at faster and faster rates. Because processor performance and memory cycle times are not keeping up with this demand, there is a fundamental need for simultaneous processing of multiple packets, and the degree of this parallelism is increasing. Sometimes a dependency exists between two packets currently being operated on, and as the ratio of packet processing time to packet transmission time i ...

Keywords: memory synchronization, multithreaded processors, network processors, packet dependencies, packet processing, parallel processing, processor architecture, thread level speculation

2 Elements of functional and performance analysis: Enabling scheduling analysis of heterogeneous systems with multi-rate data dependencies and rate intervals

Marek Jersak, Rolf Ernst

June 2003 **Proceedings of the 40th conference on Design automation**

Full text available: pdf(118.62 KB) Additional Information: full citation, abstract, references, index terms

Formal methods are growing in importance for performance analysis of real-time systems, but embedded system heterogeneity limits the application of these methods to subsystems or special cases. One of the problems is the rich variety of interactions between embedded system processes, which cannot be directly expressed with the typical event models used in real-time analysis.This paper shows how to transform complex interaction patterns into the integral representation of minimum and maximum arri ...

Keywords: arrival curves, event models, heterogeneous embedded systems, multi-rate data dependencies, multiple activating inputs, rate intervals, real-time systems, scheduling analysis

3 Unifying tool, data and process flow management

Michael Rumsey, Colin Farquhar

November 1992 **Proceedings of the conference on European design automation**

Full text available: pdf(809.45 KB) Additional Information: full citation, references, citings, index terms

4 Dependency sequences and hierarchical clocks: efficient alternatives to vector clocks for mobile computing systems

Ravi Prakash, Mukesh Singhal

October 1997 **Wireless Networks**, Volume 3 Issue 5

Full text available: pdf(260.78 KB) Additional Information: full citation, abstract, references, index terms

Vector clocks have been used to capture causal dependencies between processes in distributed computing systems. Vector clocks are not suitable for mobile computing systems due to (i) lack of scalability: its size is equal to the number of nodes, and (ii) its inability to cope with fluctuations in the number of nodes. This paper presents two efficient alternatives to vector clock, namely, sets of dependency sequences, and hierarchical clock. Both the alternatives are scalable and are immune ...

5 Special session on reconfigurable computing: Adaptive architectures for an OTN processor: reducing design costs through reconfigurability and multiprocessing

Tudor Murgan, Mihail Petrov, Mateusz Majer, Peter Zipf, Manfred Glesner, Ulrich Heinkel, Joerg Pleickhardt, Bernd Bleisteiner

April 2004 **Proceedings of the first conference on computing frontiers on Computing frontiers**

Full text available: pdf(1.01 MB) Additional Information: full citation, abstract, references, citings, index terms

The standardisation process of Optical Transport Networks generally spans a long period of time. For providers intending to be present early on the market, this implies costly design re-spins if the wrong "flavour" of the protocol standard has been implemented. Extending a protocol processing device through application specific reconfigurable elements or multiprocessor units augment its flexibility. Thus, the architecture can be upgraded to standard updates or changes not even considered at desi ...

Keywords: ITU-T G.709, multiprocessor and reconfigurable architectures, optical transport networks, standard upgrades

6 Time weaver: a software-through-models framework for embedded real-time systems

Dionisio de Niz, Raj Rajkumar

June 2003 **ACM SIGPLAN Notices , Proceedings of the 2003 ACM SIGPLAN conference on Language, compiler, and tool for embedded systems**, Volume 38 Issue 7

Full text available: pdf(467.76 KB) Additional Information: full citation, abstract, references, index terms

Embedded real-time systems are deployed in a wide range of application domains including transportation systems, automated manufacturing, process control, defense, aerospace, and telecommunications. These systems must satisfy not only logical functional requirements but also *para-functional* properties such as timeliness, Quality of Service (QoS) and reliability. The cross-cutting behaviors imposed by these para-functional properties and dependencies on operational characteristics (e.g. ha ...

Keywords: couplers, embedded, real-time, semantic dimension, semantic separation, software-through-models

7 A fast string-matching algorithm for network processor-based intrusion detection system

Rong-Tai Liu, Nen-Fu Huang, Chih-Hao Chen, Chia-Nan Kao

August 2004 **ACM Transactions on Embedded Computing Systems (TECS)**, Volume 3 Issue 3

Full text available: pdf(571.00 KB) Additional Information: full citation, abstract, references, index terms

Network intrusion detection systems (NIDSs) are one of the latest developments in security. The matching of packet strings against collected signatures dominates signature-based NIDS performance. Network processors are also one of the fastest growing segments of the semiconductor market, because they are designed to provide scalable and flexible solutions that can accommodate change quickly and economically. This work presents a fast string-matching algorithm (called FNP) over the network proces ...

Keywords: Intrusion detection, network, pattern matching, processor

8 Compressing MIPS code by multiple operand dependencies

Kelvin Lin, Chung-Ping Chung, Jean Jyh-Jiun Shann

November 2003 **ACM Transactions on Embedded Computing Systems (TECS)**, Volume 2 Issue 4

Full text available: pdf(576.31 KB) Additional Information: full citation, abstract, references, index terms

Intuitively, destination registers of some instructions have great possibilities to be used as the source

registers of the immediately subsequent instructions. Such destination register/source register pairs have been exploited previously to improve code compression ratio [*compression ratio = (Dictionary Size + Encoded Program Size)/Original Program Size*]. This paper further examines the exploitation of both register and immediate operand dependencies to improve the c ...

Keywords: Code compression, benchmarks, data compression, instruction set architecture

9 A survey of processors with explicit multithreading

Theo Ungerer, Borut Robič, Jurij Šilc

March 2003 **ACM Computing Surveys (CSUR)**, Volume 35 Issue 1

Full text available: pdf(920.16 KB) Additional Information: full citation, abstract, references, index terms

Hardware multithreading is becoming a generally applied technique in the next generation of microprocessors. Several multithreaded processors are announced by industry or already into production in the areas of high-performance microprocessors, media, and network processors.A multithreaded processor is able to pursue two or more threads of control in parallel within the processor pipeline. The contexts of two or more threads of control are often stored in separate on-chip register sets. Unused i ...

Keywords: Blocked multithreading, interleaved multithreading, simultaneous multithreading

10 Modeling dependencies in protein-DNA binding sites

Yoseph Barash, Gal Elidan, Nir Friedman, Tommy Kaplan

April 2003 **Proceedings of the seventh annual international conference on Computational molecular biology**

Full text available: pdf(411.94 KB) Additional Information: full citation, abstract, references, citings, index terms

The availability of whole genome sequences and high-throughput genomic assays opens the door for *in silico* analysis of transcription regulation. This includes methods for discovering and characterizing the binding sites of DNA-binding proteins, such as transcription factors. A common representation of transcription factor binding sites is a *position specific score matrix* (PSSM). This representation makes the strong assumption that binding site positions are independent of each othe ...

Keywords: DNA sequence motifs, bayesian networks, factors binding sites, transcription

11 Dynamic functional dependencies and database aging

Victor Vianu

January 1987 **Journal of the ACM (JACM)**, Volume 34 Issue 1

Full text available: pdf(2.35 MB) Additional Information: full citation, abstract, references, citings, index terms, review

A simple extension of the relational model is introduced to study the effects of dynamic constraints on database evolution. Both static and dynamic constraints are used in conjunction with the model. The static constraints considered here are functional dependencies (FDs). The dynamic constraints involve global updates and are restricted to certain analogs of FDs, called "dynamic" FDs. The results concern the effect of the dynamic constraints on the static constraints satisfied ...

12 Linguistic processing using a dependency structure grammar: for speech recognition and understanding

Sho-ichi Matsunaga, Masaki Kohda

August 1988 **Proceedings of the 12th conference on Computational linguistics - Volume 1**

Full text available: pdf(557.15 KB) Additional Information: full citation, abstract, references, citings

This paper proposes an efficient linguistic processing strategy for speech recognition and understanding using a dependency structure grammar. The strategy includes parsing and phrase prediction algorithms. After speech processing and phrase recognition based on phoneme recognition, the parser extracts the sentence with the best likelihood taking account of the phonetic likelihood of phrase candidates and the linguistic likelihood of the semantic inter-phrase dependency relationships. A fast par ...

13 Computing covers for embedded functional dependencies

G. Gottlob

June 1987 **Proceedings of the sixth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems**

Full text available: pdf(1.06 MB) Additional Information: full citation, abstract, references, citings, index terms

This paper deals with the problem of computing covers for the functional dependencies embedded in a subset of a given relation schema. We show how this problem can be simplified and present a new and efficient algorithm "Reduction. By Resolution" (RBR) for its solution. Though the problem of computing covers for embedded dependencies is inherently exponential, our algorithm behaves polynomially for several classes of inputs. RBR can be used for the solution of some related probl ...

14 A stochastic language model using dependency and its improvement by word clustering

Shinsuke Mori, Makoto Nagao

August 1998

Full text available: pdf(550.06 KB) Additional Information: full citation, abstract, references, citings

In this paper, we present a stochastic language model for Japanese using dependency. The prediction unit in this model is an attribute of "bunsetsu". This is represented by the product of the head of content words and that of function words. The relation between the attributes of "bunsetsu" is ruled by a context-free grammar. The word sequences are predicted from the attribute using word n-gram model. The spell of Unknow word is predicted using character n-gram model. This model is robust in tha ...

15 Session 2: Inference rules for functional and inclusion dependencies

John C. Mitchell

March 1983 **Proceedings of the 2nd ACM SIGACT-SIGMOD symposium on Principles of database systems**

Full text available: pdf(983.90 KB) Additional Information: full citation, abstract, references, citings

A set Σ of functional dependencies and inclusion dependencies *implies* a single dependency σ if all databases (finite and infinite) which satisfy Σ also satisfy σ. This paper presents complete inference rules for deducing implications of inclusion and functional dependencies. The results of [5] suggest that the implication problem for functional and inclusion dependencies together has no simple axiomatization satisfying a natural set of conditions. Out of necessity, ...

16 Session 7: Inclusion dependencies and their interaction with functional dependencies

Marco A. Casanova, Ronald Fagin, Christos H. Papadimitriou

March 1982 **Proceedings of the 1st ACM SIGACT-SIGMOD symposium on Principles of database systems**

Full text available: pdf(669.68 KB) Additional Information: full citation, abstract, references, citings

Inclusion dependencies, or INDs (which can say, for example, that every manager is an employee) are studied, including their interaction with functional dependencies, or FDs. A simple complete axiomatization for INDs is presented, and the decision problem for INDs is shown to be PSPACE-complete. (The decision problem for INDs is the problem of determining whether or not Σ logically implies σ, given a set Σ of INDs and a single IND σ). It is shown that finite implication (...

Keywords: PSPACE-complete, complete axiomatization, functional dependency, inclusion dependency, relational database

17 Implementation and evaluation of a QoS-capable cluster-based IP router

Prashant Pradhan, Tzi-cker Chiueh

November 2002 **Proceedings of the 2002 ACM/IEEE conference on Supercomputing**

Full text available: pdf(215.68 KB) Additional Information: full citation, abstract, references, index terms

A major challenge in Internet edge router design is to support both high packet forwarding performance and versatile and efficient packet processing capabilities. The thesis of this research project is that a cluster of PCs connected by a high speed system area network provides an effective hardware platform for building routers to be used at the edges of the Internet. This paper describes a scalable and extensible edge router architecture called *Panama*, which supports a novel aggregate r ...

18 Formal aspects and parsing issues of dependency theory

Vincenzo Lombardo, Leonardo Lesmo

August 1998

Full text available: pdf(679.37 KB) Publisher Site Additional Information: full citation, abstract, references

The paper investigates the problem of providing a formal device for the dependency approach to syntax, and to link it with a parsing model. After reviewing the basic tenets of the paradigm and the few existing mathematical results, we describe a dependency formalism which is able to deal with long-distance dependencies. Finally, we present an Earley-style parser for the formalism and discuss the (polynomial) complexity results.

19 Novel design methodologies and signal integrity: Design of a 17-million gate network processor using a design factory

Gilles-Eric Descamps, Satish Bagalkotkar, Subramanian Ganesan, Satish Iyengar, Alain Pirson

June 2003 **Proceedings of the 40th conference on Design automation**

Full text available: pdf(1.75 MB) Additional Information: full citation, abstract, references, index terms

Silicon Access Networks taped out in one year four high performance SoC products: a high-end Network Processor and three associated Co-processors, providing the industry with the highest performance OC-192 Data Plane Processing solution. The four chips are shipping for revenue and went into production from first silicon with no mask change. They were designed using state-of-the-art 0.13µm technology and collectively represent about 750-million transistors, implementing a variety of analog, ...

20 Dependency parsing with an extended finite state approach

Kemal Oflazer

June 1999 **Proceedings of the 37th conference on Association for Computational Linguistics**

Full text available: pdf(683.26 KB) Additional Information: full citation, abstract, references

This paper presents a dependency parsing scheme using an extended finite state approach. The parser augments input representation with "channels" so that links representing syntactic dependency relations among words can be accommodated, and iterates on the input a number of times to arrive at a fixed point. Intermediate configurations violating various constraints of projective dependency representations such as no crossing links, no independent items except sentential head, etc, are filtered ...

http://portal.acm.org/results.cfm?coll=ACM&dl=ACM&CFID=35858585&CFTOKEN=16765106

Terms used network processor dependency or dependencies sequence

Found 15,074 of 148,786

Sort results by relevance

Display results expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 21 - 40 of 200 Result page: previous 1 2 3 4 5 6 7 8 9 10 next

Best 200 shown

Relevance scale

21 Testing implications of data dependencies

David Maier, Alberto O. Mendelzon, Yehoshua Sagiv

December 1979 **ACM Transactions on Database Systems (TODS)**, Volume 4 Issue 4

Full text available: pdf(1.14 MB) Additional Information: full citation, abstract, references, citings, index terms

Presented is a computation method—the chase—for testing implication of data dependencies by a set of data dependencies. The chase operates on tableaux similar to those of Aho, Sagiv, and Ullman. The chase includes previous tableau computation methods as special cases. By interpreting tableaux alternately as mappings or as templates for relations, it is possible to test implication of join dependencies (including multivalued dependencies) and functional dependenc ...

Keywords: chase, data dependencies, functional dependencies, join dependencies, multivalued dependencies, relational databases, tableaux

22 Finitely Specifiable Implicational Dependency Families

Richard Hull

March 1984 **Journal of the ACM (JACM)**, Volume 31 Issue 2

Full text available: pdf(958.88 KB) Additional Information: full citation, references, citings, index terms, review

23 Dependency maintenance in declarative geometry modelling

Rüdiger Klein

May 1997 **Proceedings of the fourth ACM symposium on Solid modeling and applications**

Full text available: pdf(1.16 MB) Additional Information: full citation, references, index terms

24 Embedded implicational dependencies and their inference problem

Ashok K. Chandra, Harry R. Lewis, Johann A. Makowsky

May 1981 **Proceedings of the thirteenth annual ACM symposium on Theory of computing**

Full text available: pdf(939.79 KB) Additional Information: full citation, abstract, references, citings, index terms

It is shown that the general inference problem for embedded implicational dependencies (EIDs) is undecidable. For the more important case of finite inference (i.e., inference for finite data bases), the problem is not even recursively enumerable (r.e.); rather, it is complete in co-r.e. These results hold even for typed EIDs without equality, as well as for (untyped) template dependencies. The case for typed template dependencies remains open. The complexity of the inference problem for ful ...

25 Reasoning about functional dependencies generalized for semantic data models

Grant E. Weddell

March 1992

ACM Transactions on Database Systems (TODS), Volume 17 Issue 1

Full text available: pdf(2.01 MB) Additional Information: full citation, abstract, references, citings, index terms, review

We propose a more general form of functional dependency for semantic data models that derives from their common feature in which the separate notions of domain and relation in the relational model are combined into a single notion of class. This usually results in a richer terminological component for their query languages, whereby terms may navigate through any number of properties, including none. We prove the richer expressiveness of thi ...

Keywords: constraint theory, functional dependencies, query optimization, semantic data models

26 Computational complexity and grammatical formalisms: Crossed serial dependencies: a low-power parseable extension to GPSG

Henry Thompson

June 1983 **Proceedings of the 21st conference on Association for Computational Linguistics**

Full text available: pdf(390.56 KB) Publisher Site Additional Information: full citation, abstract, references

An extension to the GPSG grammatical formalism is proposed, allowing non-terminals to consist of finite sequences of category labels, and allowing schematic variables to range over such sequences. The extension is shown to be sufficient to provide a strongly adequate grammar for crossed serial dependencies, as found in e.g. Dutch subordinate clauses. The structures induced for such constructions are argued to be more appropriate to data involving conjunction than some previous proposals have bee ...

27 Data dependency theories: The interaction between functional dependencies and template dependencies

Fereidoon Sadri, Jeffrey D. Ullman

May 1980 **Proceedings of the 1980 ACM SIGMOD international conference on Management of data**

Full text available: pdf(682.63 KB) Additional Information: full citation, abstract, references, citings

A large class of dependencies, called template dependencies, was introduced in Sadri and Ullman [1979], and a complete set of inference rules (axioms) was given for it. In this paper, we investigate the interaction between template dependencies and functional dependencies. We develop techniques for partially deciding which template and functional dependencies are logically implied by a set of template and functional dependencies. Since template dependencies include all dependencies, besides func ...

28 Notions of dependency satisfaction

Marc H. Graham, Alberto O. Mendelzon, Moshe Y. Vardi

January 1986 **Journal of the ACM (JACM)**, Volume 33 Issue 1

Full text available: pdf(1.83 MB) Additional Information: full citation, abstract, references, citings, index terms, review

Two notions of dependency satisfaction, consistency and completeness, are introduced. Consistency is the natural generalization of weak-instance satisfaction and seems appropriate when only equality-generating dependencies are given, but disagrees with the standard notion in the presence of tuple-generating dependencies. Completeness is based on the intuitive semantics of tuple-generating dependencies but differs from the standard notion for equality-genera ...

29 Rollback-dependency trackability: visible characterizations

Roberto Baldoni, Jean-Michel Hélary, Michel Raynal

May 1999 **Proceedings of the eighteenth annual ACM symposium on Principles of distributed computing**

Full text available: pdf(938.66 KB) Additional Information: full citation, references, index terms

30 Session 2: embedded system techniques (1): A near-optimal instruction scheduler for a tightly constrained, variable instruction set embedded processor

Jack Liu, Fred Chow

October 2002 **Proceedings of the 2002 international conference on Compilers, architecture, and synthesis for embedded systems**

Full text available: pdf(202.21 KB) Additional Information: full citation, abstract, references, index terms

This paper describes the implementation of an instruction scheduler that produces near-optimal results by efficiently enumerating all possible schedules. The scheduler is implemented in the context of an embedded processor that allows compile-time configuration of the instruction set to use for each program, where the need to adhere to the tight and irregular hardware constraints were the original motivation for using this approach. We present techniques that speed up the enumeration process by ...

Keywords: configurable code generation, dictionary, embedded processor, enumeration, instruction scheduling, program representation, resource modeling, variable instruction set

31 Compilation: Efficient spill code for SDRAM

V. Krishna Nandivada, Jens Palsberg

October 2003 **Proceedings of the 2003 international conference on Compilers, architectures and synthesis for embedded systems**

Full text available: pdf(199.32 KB) Additional Information: full citation, abstract, references, index terms

Processors such as StrongARM and memory such as SDRAM enable efficient execution of multiple loads and stores in a single instruction. This is particularly useful in connection with register allocation where spill code may need to save and restore multiple registers. Until now, there has been no effective strategy for utilizing this to its full potential. In this paper we investigate the use of SDRAM for optimization of spill code. The core of the problem is to arrange the variables in the spill ...

Keywords: SDRAM, integer linear programming, memory layout, optimization

32 Timing analysis and memory optimization for embedded systems: Schedulability of event-driven code blocks in real-time embedded systems

Samarjit Chakraborty, Thomas Erlebach, Simon Künzli, Lothar Thiele

June 2002 **Proceedings of the 39th conference on Design automation**

Full text available: pdf(280.31 KB) Additional Information: full citation, abstract, references, index terms

Many real-time embedded systems involve a collection of independently executing event-driven code blocks, having hard real-time constraints. Tasks in many such systems, like network processors, are either not preemptable or have restrictions on the number of preemptions allowed. All the previous work on the schedulability analysis of such systems either have exponential complexity, or allow unbounded number of preemptions and are usually based on heuristics. In this paper we present the exact ne ...

33 Learning dependency translation models as collections of finite-state head transducers

Hiyan Alshawi, Shona Douglas, Srinivas Bangalore

March 2000 **Computational Linguistics**, Volume 26 Issue 1

Full text available: pdf(1.00 MB) Publisher Site Additional Information: full citation, abstract, references

The paper defines weighted head transducers, finite-state machines that perform middle-out string transduction. These transducers are strictly more expressive than the special case of standard left-to-right finite-state transducers. Dependency transduction models are then defined as collections of weighted head transducers that are applied hierarchically. A dynamic programming search algorithm is described for finding the optimal transduction of an input string with respect to a dependency trans ...

34 Hardware Scheduling for Dynamic Adaptability using External Profiling and Hardware Threading

Brian Swahn, Soha Hassoun

November 2003 **Proceedings of the 2003 IEEE/ACM international conference on Computer-aided design**

Full text available: pdf(318.89 KB) Additional Information: full citation, abstract, index terms

While performance, area, and power constraints have been thedriving force in designing current communication-enabled embeddedsystems, post-fabrication and run-time adaptability is now required.Two dominant configurable hardware platforms are processorsand FPGAs. However, for compute-intensive applications,neither platform delivers the needed performance at the desiredlow power. The need thus arises for custom, application-specificconfigurable (ASC) hardware.This paper addresses the optimization ...

35

A program form based on data dependency in predicate regions
Jeanne Ferrante, Karl J. Ottenstein
January 1983 **Proceedings of the 10th ACM SIGACT-SIGPLAN symposium on Principles of programming languages**
Full text available: pdf(1.15 MB) Additional Information: full citation, abstract, references, citings

A new program representation is presented which permits certain optimizations to be performed at less expense than with other forms. This paper describes code motion, common subexpression elimination and induction variable detection. Scalar propagation and constant folding are sketched here, but detailed elsewhere. The powerful code motion strategy allows entire regions of the program to be moved. The representation described may be used as a compiler intermediate form or simply as a model for p ...

36 Session: formal topics: First order logic formalization for functional, multivalued and mutual dependencies
J. M. Nicolas
May 1978 **Proceedings of the 1978 ACM SIGMOD international conference on management of data**
Full text available: pdf(917.36 KB) Additional Information: full citation, abstract, references, citings

The purpose of this paper is to show that first order logic is adequate for formalizing functional, multivalued and mutual dependencies in relational data bases. Advantages of using logic instead of tailored formal systems are presented. This paper is decomposed into four sections. The first one presents some notions of logic and theorem proving which are relevant to this study. In the second section, a way to analyze data bases in terms of logic is briefly indicated. The third section deals wit ...

Keywords: functional dependencies, logic and data bases, multivalued dependencies, mutual dependencies, relational data base

37 Object normal forms and dependency constraints for object-oriented schemata
Zahir Tari, John Stokes, Stefano Spaccapietra
December 1997 **ACM Transactions on Database Systems (TODS)**, Volume 22 Issue 4
Full text available: pdf(439.37 KB) Additional Information: full citation, abstract, references, citings, index terms, review

We address the development of a normalization theory for object-oriented data models that have common features to support objects. We first provide an extension of functional dependencies to cope with the richer semantics of relationships between objects, called path dependency, local dependency, and global dependency constraints. Using these dependency constraints, we provide normal forms for object-oriented data models based on the notions of user ...

Keywords: data model, functional and multivalued dependencies, normal forms, object-oriented paradigm

38 Regular papers: Acquisition of phrase-level bilingual correspondence using dependency structure
Kaoru Yamamoto, Yuji Matsumoto
July 2000 **Proceedings of the 17th conference on Computational linguistics - Volume 2**
Full text available: pdf(600.45 KB) Additional Information: full citation, abstract, references

This paper describes a method to find phrase-level translation patterns from parallel corpora by applying dependency structure analysis. We use statistical dependency parsers to determine dependency relations between base phrases in a sentence. Our method is tested with a business expression corpus containing 10000 English-Japanese sentence pairs and achieved approximately 90% accuracy in extracting bilingual correspondences. The result shows that the use of dependency relation helps to acquire ...

39 Separating surface order and syntactic relations in a dependency grammar
Norbert Bröker
August 1998
Full text available: pdf(654.27 KB) Publisher Site Additional Information: full citation, abstract, references

This paper proposes decoupling the dependency tree from word order, such that surface ordering is not determined by traversing the dependency tree. We develop the notion of a *word order domain structure*,

which is linked but structurally dissimilar to the syntactic dependency tree. The proposal results in a lexicalized, declarative, and formally precise description of word order; features which lack previous proposals for dependency grammars. Contrary to other lexicalized approaches to wor ...

40 Horn clauses and database dependencies (Extended Abstract)

Ronald Fagin

April 1980 **Proceedings of the twelfth annual ACM symposium on Theory of computing**

Full text available: pdf(1.09 MB) Additional Information: full citation, abstract, references, citings, index terms

In the last year or so, a number of generalizations of these dependencies have appeared: Nicolas's mutual dependencies [Ni], which say that a relation is the join of three of its projections; Rissanen's and Aho, Beeri, and Ullman's join dependencies ([Ri], [ABU]), which generalize further to an arbitrary number of projections; Paradaens' transitive dependencies [Pa], which generalize both FDs and MVDs; Sagiv and Walecka's subset dep ...

Terms used **network processor dependency** or **dependencies sequence**

Found **15,074** of **148,786**

Sort results by relevance

Display results expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 41 - 60 of 200 Result page: previous 1 2 **3** 4 5 6 7 8 9 10 next

Best 200 shown

Relevance scale

41 Graphics Programming Using a Database System with Dependency Declarations

M. T. Garrett, J. D. Foley

April 1982 **ACM Transactions on Graphics (TOG)**, Volume 1 Issue 2

Full text available: pdf(1.26 MB) Additional Information: full citation, references, citings, index terms

Keywords: dependencies, nonprocedural programming, production rules

42 The complexity of recognition of linguistically adequate dependency grammars

Peter Neuhaus, Norbert Bröker

July 1997

Full text available: pdf(612.31 KB) Publisher Site Additional Information: full citation, abstract, references, citings

Results of computational complexity exist for a wide range of phrase structure-based grammar formalisms, while there is an apparent lack of such results for dependency-based formalisms. We here adapt a result on the complexity of ID/LP-grammars to the dependency framework. Contrary to previous studies on heavily restricted dependency grammars, we prove that recognition (and thus, parsing) of linguistically adequate dependency grammars is *NP*-complete.

43 Dependence flow graphs: an algebraic approach to program dependencies

Keshav Pingali, Micah Beck, Richard Johnson, Mayan Moudgill, Paul Stodghill

January 1991 **Proceedings of the 18th ACM SIGPLAN-SIGACT symposium on Principles of programming languages**

Full text available: pdf(1.30 MB) Additional Information: full citation, references, citings, index terms

44 Tuple sequences and lexicographic indexes

Serge Abiteboul, Seymour Ginsburg

May 1986 **Journal of the ACM (JACM)**, Volume 33 Issue 3

Full text available: pdf(981.91 KB) Additional Information: full citation, abstract, references, citings, index terms, review

The concept of a tuple sequence is introduced in order to investigate structure connected with relational model implementation. Analogs are presented for the relational operations of projection, join, and selection, and the decomposition problem for tuple sequences is considered. The lexicographical ordering of tuple sequences is studied via the notion of (lexicographic) index. A sound and complete set of inference rules for indexes is exhibited, and two algorithmic questions related to ind ...

45 Papers: Cross-serial dependencies are not hard to process

Carl Vogel, Ulrike Hahn, Holly Branigan
August 1996 **Proceedings of the 16th conference on Computational linguistics - Volume 1**

Full text available: pdf(665.87 KB) Additional Information: full citation, abstract, references

Cross-serial dependencies in Dutch and Swiss-German are the only known extracontext free natural language syntactic phenomena. Psycholinguistic evidence suggests cross-serial orderings tend to be easier to process than nested constructions. We argue that the expressivity requirements of the corresponding formal languages do not actually entail that processing reduplication languages require the worst-case time complexity for languages of the same expressive class. We distinguish between context- ...

46 Testing satisfaction of functional dependencies
Peter Honeyman
July 1982 **Journal of the ACM (JACM)**, Volume 29 Issue 3

Full text available: pdf(517.96 KB) Additional Information: full citation, references, citings, index terms

47 An Algorithm for Inferring Multivalued Dependencies with an Application to Propositional Logic
Yehoshua Sagiv
April 1980 **Journal of the ACM (JACM)**, Volume 27 Issue 2

Full text available: pdf(891.74 KB) Additional Information: full citation, abstract, references, citings, index terms

An algorithm is given for deciding whether a functional or a multivalued dependency &sgr; (with a right-hand side Y) is implied by a set of functional and multivalued dependencies &Sgr;. The running time of the algorithm is O(| Y |‖&Sgr;‖), where Y is the number of attributes in Y and ‖&Sgr;‖ is the size of the description of &Sgr;. The problem of constructing th ...

48 On the menbership problem for functional and multivalued dependencies in relational databases
Catriel Beeri
September 1980 **ACM Transactions on Database Systems (TODS)**, Volume 5 Issue 3

Full text available: pdf(1.51 MB) Additional Information: full citation, abstract, references, citings, index terms

The problem of whether a given dependency in a database relation can be derived from a given set of dependencies is investigated. We show that the problem can be decided in polynomial time when the given set consists of either multivalued dependencies only or of both functional and multivalued dependencies and the given dependency is also either a functional or a multivalued dependency. These results hold when the derivations are restricted not to use the complementation rule.

Keywords: functional dependency, inference rule, membership, multivalued dependency, relations

49 Dependency analysis of Ada programs
Janusz Laski, William Stanley, Jim Hurst
November 1998 **ACM SIGAda Ada Letters , Proceedings of the 1998 annual ACM SIGAda international conference on Ada**, Volume XVIII Issue 6

Full text available: pdf(1.39 MB) Additional Information: full citation, references, citings, index terms

Keywords: Ada, information flow, path analysis, program anomalies, program dependencies, static analysis

50 Abstracting dependencies between software configuration items
Carl A. Gunter
January 2000 **ACM Transactions on Software Engineering and Methodology (TOSEM)**, Volume 9 Issue 1

Full text available: pdf(271.64 KB) Additional Information: full citation, abstract, references, index terms

This article studies an abstract model of dependencies between software configuration items based on a theory of concurrent computation over a class of Petri nets called production nets. A general theory of build optimizations and their correctness is developed based on a form of abstract interpretation called a build abstraction; these are created during a build and are used to optimize subsequent builds. Various examples of such optimizations are discusse ...

Keywords: Petri nets, abstract interpretation, mathematical models of build dependencies, software configuration management

51 Reconfiguration of spare capacity for MPLS-based recovery in the internet backbone networks
Pin-Han Ho, Hussein T. Mouftah
February 2004 **IEEE/ACM Transactions on Networking (TON)**, Volume 12 Issue 1
Full text available: pdf(578.55 KB) Additional Information: full citation, abstract, references, index terms

This paper introduces a novel approach, called Short Leap Shared Protection with spare capacity Reallocation (SLSP-R), to deal with dynamic reconfiguration of spare capacity for MPLS-based recovery in the Internet backbone networks. SLSP-R is based on the SLSP framework and is designed to quantify the impact of computation complexity on network performance. The basic idea for SLSP-R is to subdivide a lengthy optimization process into several subtasks in order to trade the optimization quality wi ...

Keywords: MPLS, network planning, ring-based architecture, spare capacity, survivability

52 Session 1: Dependency characterizations for acyclic database schemes
Gösta Grahne, Kari Jouko Räihä
April 1984 **Proceedings of the 3rd ACM SIGACT-SIGMOD symposium on Principles of database systems**
Full text available: pdf(609.05 KB) Additional Information: full citation, abstract, references

Acyclic database schemes have attracted a lot of interest because of the nice properties enjoyed by such schemes. Recently some new acyclicity conditions that are strictly stronger than the normal α-acyclicity have been introduced by Fagin. Because of increased requirements, the database schemes in the new classes have some further useful properties that are not shared by α-acyclic schemes. Therefore the new classes have practical relevance.A database designer may work in terms of at ...

53 A polynomial-order algorithm for optimal phrase sequence selection from a phrase lattice and its parallel layered implementation
Kazuhiko Ozeki
August 1990 **Proceedings of the 13th conference on Computational linguistics - Volume 2**
Full text available: pdf(497.71 KB) Additional Information: full citation, abstract, references

This paper deals with a problem of selecting an optimal phrase sequence from a phrase lattice, which is often encountered in language processing such as word processing and post-processing for speech recognition. The problem is formulated as one of combinatorial optimization and a polynomial order algorithm is derived. This algorithm finds an optimal phrase sequence and its dependency structure simultaneously, and is therefore particularly suited for an interface between speech recognition and v ...

54 On the family of generalized dependency constraints
John Grant, Barry E. Jacobs
October 1982 **Journal of the ACM (JACM)**, Volume 29 Issue 4
Full text available: pdf(677.44 KB) Additional Information: full citation, references, citings, index terms

55 Subset Dependencies and a Completeness Result for a Subclass of Embedded Multivalued Dependencies
Yehoshua Sagiv, Scott F. Walecka
January 1982 **Journal of the ACM (JACM)**, Volume 29 Issue 1
Full text available: pdf(803.23 KB) Additional Information: full citation, references, citings, index terms

56 Physical database design: Applying approximate order dependency to reduce indexing space
Jirun Dong, Richard Hull
June 1982 **Proceedings of the 1982 ACM SIGMOD international conference on Management of data**

Full text available: pdf(1.22 MB) Additional Information: full citation, abstract, references, citings

The recently introduced notion of order dependency in the relational model is generalized to include situations where order dependency is satisfied in an approximate way. Two fundamental types of approximate satisfaction are distinguished and analyzed. It is shown for both types that such approximate satisfaction of order dependencies can be applied to substantially reduce indexing space without significantly increasing access time.

57 Determining View dependencies using tableaux

Anthony Klug, Rod Price

September 1982 **ACM Transactions on Database Systems (TODS)**, Volume 7 Issue 3

Full text available: pdf(1.37 MB) Additional Information: full citation, abstract, references, citings, index terms

A relational database models some part of the real world by a set of relations and a set of constraints. The constraints model properties of the stored information and must be maintained true at all times. For views defined over physically stored (base) relations, this is done by determining whether the view constraints are logical consequences of base relation constraints. A technique for determining such valid view constraints is presented in this paper. A generalization of the tableau ch ...

Keywords: chase, dependencies, rational algebra, relational model, tableaux

58 Synthesizing third normal form relations from functional dependencies

Philip A. Bernstein

December 1976 **ACM Transactions on Database Systems (TODS)**, Volume 1 Issue 4

Full text available: pdf(1.79 MB) Additional Information: full citation, abstract, references, citings, index terms

It has been proposed that the description of a relational database can be formulated as a set of functional relationships among database attributes. These functional relationships can then be used to synthesize algorithmically a relational scheme. It is the purpose of this paper to present an effective procedure for performing such a synthesis. The schema that results from this procedure is proved to be in Codd's third normal form and to contain the fewest possible number of relations. Prob ...

Keywords: database schema, functional dependency, relational model, semantics of data, third normal form

59 Optimal online scheduling of parallel jobs with dependencies

Anja Feldmann, Ming-Yang Kao, Jiří Sgall, Shang-Hua Teng

June 1993 **Proceedings of the twenty-fifth annual ACM symposium on Theory of computing**

Full text available: pdf(1.09 MB) Additional Information: full citation, references, citings, index terms

60 Design of a dependency-directed compiler for constraint propagation

Roy Feldman

June 1988 **Proceedings of the first international conference on Industrial and engineering applications of artificial intelligence and expert systems - Volume 1**

Full text available: pdf(467.44 KB) Additional Information: full citation, abstract, references, index terms

This paper describes COCO (COnstraint COmpiler), an optimizing compiler for a constraint language embedded within a frame system. The approach to constraint optimization is based on the following observation: when there are multiple inputs to a constraint network the order in which changes are propagated affects overall computational cost. Any uniform algorithm for propagating values through a network that does not take into account data dependencies will produce sub-optimal results. COCO p ...

http://portal.acm.org/results.cfm?query=%22network%20processor%22%20%28dependency%20or%20dependencie...

Terms used **network processor sequence number**

Found **1,196** of **148,786**

Sort results by relevance

Display results expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown

Relevance scale

1 Session 7: embedded system techniques (2): Handling of packet dependencies: a critical issue for highly parallel network processors

Stephen Melvin, Yale Patt

October 2002 **Proceedings of the 2002 international conference on Compilers, architecture, and synthesis for embedded systems**

Full text available: pdf(221.66 KB) Additional Information: full citation, abstract, references, citings, index terms

Network processors are being asked to perform increasingly complex operations on packets of information at faster and faster rates. Because processor performance and memory cycle times are not keeping up with this demand, there is a fundamental need for simultaneous processing of multiple packets, and the degree of this parallelism is increasing. Sometimes a dependency exists between two packets currently being operated on, and as the ratio of packet processing time to packet transmission time i ...

Keywords: memory synchronization, multithreaded processors, network processors, packet dependencies, packet processing, parallel processing, processor architecture, thread level speculation

2 Lightweight network support for scalable end-to-end services

Kenneth L. Calvert, James Griffioen, Su Wen

August 2002 **ACM SIGCOMM Computer Communication Review , Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications**, Volume 32 Issue 4

Full text available: pdf(331.84 KB) Additional Information: full citation, abstract, references, citings, index terms

Some end-to-end network services benefit greatly from network support in terms of utility and scalability. However, when such support is provided through service-specific mechanisms, the proliferation of one-off solutions tend to decrease the robustness of the network over time. Programmable routers, on the other hand, offer generic support for a variety of end-to-end services, but face a different set of challenges with respect to performance, scalability, security, and robustness. Ideally, rou ...

Keywords: end-to-end services, ephemeral state, programmable network, router achitecture

3 Simulation via implementation with applications in computer communication

Kenneth Brayer, Valerie Lafleur, Gary Simpson

March 1982 **Proceedings of the 15th annual symposium on Simulation**

Full text available: pdf(1.39 MB) Additional Information: full citation, abstract, index terms

The traditional approach to performing discrete digital simulation has been one of developing a mathematical or statistical model to represent a process, programming this model on a large scale computer, and then executing the model to obtain performance results. In this study, the authors have developed a simulation of a computer communication network by simulating the users in a central computer and implementing the remainder of the network in actual network processors. This allows for au ...

4 Decentralized mutual exclusion in the presence of link failures
C. S. Hsieh, R. A. Mata
February 1989 **Proceedings of the seventeenth annual ACM conference on Computer science : Computing trends in the 1990's: Computing trends in the 1990's**
Full text available: pdf(364.02 KB) Additional Information: full citation, abstract, references, index terms

A distributed algorithm for achieving mutual exclusion in a point-to-point network with lossy communication links is presented. A communication link is said to be lossy if it loses messages at will. The algorithm tolerates any number of lossy communication links, as long as the link failures do not partition the communication network. The message network is completely asynchronous and, hence, link failures are not detectable. In a network with n nodes and e ...

5 Analysis of the parallel packet switch architecture
Sundar Iyer, Nick W. McKeown
April 2003 **IEEE/ACM Transactions on Networking (TON)**, Volume 11 Issue 2
Full text available: pdf(619.44 KB) Additional Information: full citation, abstract, references, index terms

Our work is motivated by the desire to design packet switches with large aggregate capacity and fast line rates. In this paper, we consider building a packet switch from multiple lower speed packet switches operating independently and in parallel. In particular, we consider a (perhaps obvious) parallel packet switch (PPS) architecture in which arriving traffic is demultiplexed over*k* identical lower speed packet switches, switched to the correct output port, then recombined (multiplexed) b ...

Keywords: Clos network, inverse multiplexing, load balancing, output queueing, packet switch

6 Construction of a fault-tolerant distributed tuple-space
Lewis I. Patterson, Richard S. Turner, Robert M. Hyatt
March 1993 **Proceedings of the 1993 ACM/SIGAPP symposium on Applied computing: states of the art and practice**
Full text available: pdf(634.56 KB) Additional Information: full citation, references, citings, index terms

Keywords: associative memory, fault-tolerance, shared memory

7 The distributed V kernel and its performance for diskless workstations
David R. Cheriton, Willy Zwaenepoel
October 1983 **ACM SIGOPS Operating Systems Review , Proceedings of the ninth ACM symposium on Operating systems principles**, Volume 17 Issue 5
Full text available: pdf(1.09 MB) Additional Information: full citation, abstract, references, citings, index terms

The distributed V kernel is a message-oriented kernel that provides uniform local and network interprocess communication. It is primarily being used in an environment of diskless workstations connected by a high-speed local network to a set of file servers. We describe a performance evaluation of the kernel, with particular emphasis on the cost of network file access. Our results show that over a local network: 1. Diskless workstations can access remote files with minimal perform ...

8 Selecting sequence numbers
Raymond S. Tomlinson
January 1975 **ACM SIGOPS Operating Systems Review , Proceedings of the 1975 ACM SIGCOMM/SIGOPS workshop on Interprocess communications**, Volume 9 Issue 3
Full text available: pdf(645.92 KB) Additional Information: full citation, abstract, references, citings, index terms

This paper discusses techniques for selecting and synchronizing sequence numbers such that no errors will occur if certain network characteristics can be bounded and if adequate data error detection measures are taken. The discussion specifically focuses on the protocol described by Cerf and Kahn, (1) but the ideas are applicable to other similar protocols.

9 Selecting sequence numbers
Raymond S. Tomlinson
January 1995

ACM SIGCOMM Computer Communication Review, Volume 25 Issue 1

Full text available: pdf(498.56 KB) Additional Information: full citation, abstract, index terms

A characteristic of almost all communication protocols is the use of unique numbers to identify individual pieces of data. These identifiers permit error control through acknowledgement and retransmission techniques. Usually successive pieces of data are identified with sequential numbers and the identifiers are thus called sequence numbers.This paper discusses techniques for selecting and synchronizing sequence numbers such that no errors will occur if certain network characteristics can be bou ...

10 Balancing register allocation across threads for a multithreaded network processor

Xiaotong Zhuang, Santosh Pande

June 2004 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 2004 conference on Programming language design and implementation**, Volume 39 Issue 6

Full text available: pdf(429.85 KB) Additional Information: full citation, abstract, references, index terms

Modern network processors employ multi-threading to allow concurrency amongst multiple packet processing tasks. We studied the properties of applications running on the network processors and observed that their imbalanced register requirements across different threads at different program points could lead to poor performance. Many times application needs demand some threads to be more performance critical than others and thus by controlling the register allocation across threads one could impa ...

Keywords: multithreaded processor, network processor, register allocation

11 A fast string-matching algorithm for network processor-based intrusion detection system

Rong-Tai Liu, Nen-Fu Huang, Chih-Hao Chen, Chia-Nan Kao

August 2004 **ACM Transactions on Embedded Computing Systems (TECS)**, Volume 3 Issue 3

Full text available: pdf(571.00 KB) Additional Information: full citation, abstract, references, index terms

Network intrusion detection systems (NIDSs) are one of the latest developments in security. The matching of packet strings against collected signatures dominates signature-based NIDS performance. Network processors are also one of the fastest growing segments of the semiconductor market, because they are designed to provide scalable and flexible solutions that can accommodate change quickly and economically. This work presents a fast string-matching algorithm (called FNP) over the network proces ...

Keywords: Intrusion detection, network, pattern matching, processor

12 Session 1B: System-level exploration and design: NetBench: a benchmarking suite for network processors

Gokhan Memik, William H. Mangione-Smith, Wendong Hu

November 2001 **Proceedings of the 2001 IEEE/ACM international conference on Computer-aided design**

Full text available: pdf(77.08 KB) Additional Information: full citation, abstract, references, citings, index terms

In this study we introduce NetBench, a benchmarking suite for network processors. NetBench contains a total of 9 applications that are representative of commercial applications for network processors. These applications are from all levels of packet processing; Small, low-level code fragments as well as large application level programs are included in the suite.Using SimpleScalar simulator we study the NetBench programs in detail and characterize the network processor workloads. We also compare ...

13 More on selecting sequence numbers

Yogen K. Dalal

January 1975 **ACM SIGOPS Operating Systems Review , Proceedings of the 1975 ACM SIGCOMM/SIGOPS workshop on Interprocess communications**, Volume 9 Issue 3

Full text available: pdf(506.00 KB) Additional Information: full citation, abstract, references, index terms

This note serves to supplement and extend the ideas and issues raised in [TO74]. Tomlinson examines the problem of establishing a connection (association), and being able to detect delayed packets of an old incarnation of a connection, when it is being opened and closed in quick succesion, or when the connection breaks owing to some crash and is restarted later. No protocol can guarantee reliability of communication given certain types of crashes [SU74], [BE74]. This last issue is out of th ...

14 Emerging areas: Programming challenges in network processor deployment

Chidamber Kulkarni, Matthias Gries, Christian Sauer, Kurt Keutzer

October 2003 **Proceedings of the 2003 international conference on Compilers, architectures and synthesis for embedded systems**

Full text available: pdf(234.71 KB) Additional Information: full citation, abstract, references, index terms

Programming multi-processor ASIPs, such as network processors, remains an art due to the wide variety of architectures and due to little support for exploring different implementation alternatives. We present a study that implements an IP forwarding router application on two different network processors to better understand the main challenges in programming such multi-processor ASIPs. The goal of this study is to identify the elements central to a successful deployment of such systems based on ...

Keywords: IPv4 forwarding, mapping, multi-threading, programming heterogeneous architectures, programming model, resource sharing

15 A pipelined memory architecture for high throughput network processors

Timothy Sherwood, George Varghese, Brad Calder

May 2003 **ACM SIGARCH Computer Architecture News , Proceedings of the 30th annual international symposium on Computer architecture**, Volume 31 Issue 2

Full text available: pdf(213.66 KB) Additional Information: full citation, abstract, references, citings

Designing ASICs for each new generation of backbone routers is a time intensive and fiscally draining process. In this paper we focus on the design of a programmable architecture for backbone routers, based on the manipulation of wide irregular memory words, that can provide a feasible design alternative to custom ASICs. We propose a pipelined memory design that emphasizes worst-case throughput over latency, and co-explore architectural tradeoffs with the design of several important network algo ...

16 Network processors: a perspective on market requirements, processor architectures and embedded S/W tools

P. Paulin, F. Karim, P. Bromley

March 2001 **Proceedings of the conference on Design, automation and test in Europe**

Full text available: pdf(269.19 KB) Additional Information: full citation, references, citings, index terms

17 Efficient use of memory bandwidth to improve network processor throughput

Jahangir Hasan, Satish Chandra, T. N. Vijaykumar

May 2003 **ACM SIGARCH Computer Architecture News , Proceedings of the 30th annual international symposium on Computer architecture**, Volume 31 Issue 2

Full text available: pdf(184.83 KB) Additional Information: full citation, abstract, references

We consider the efficiency of packet buffers used in packet switches built using network processors (NPs). Packet buffers are typically implemented using DRAM, which provides plentiful buffering at a reasonable cost. The problem we address is that a typical NP workload may be unable to utilize the peak DRAM bandwidth. Since the bandwidth of the packet buffer is often the bottleneck in the performance of a shared-memory packet switch, inefficient use of available DRAM bandwidth further reduces th ...

18 Memory optimization: Automatic data partitioning for the agere payload plus network processor

Steve Carr, Philip Sweany

September 2004 **Proceedings of the 2004 international conference on Compilers, architecture, and synthesis for embedded systems**

Full text available: pdf(195.89 KB) Additional Information: full citation, abstract, references, index terms

With the ever-increasing pervasiveness of the Internet and its stringent performance requirements, network system designers have begun utilizing specialized chips to increase the performance of network functions. To increase performance, many more advanced functions, such as traffic shaping and policing, are being implemented at the network interface layer to reduce delays that occur when these functions are handled by a general-purpose CPU. While some designs use ASICs to handle network functio ...

Keywords: network processors, partitioning, scheduling

19 Session S4.1: power in memory and network processors: Increasing power efficiency of multi-core network processors through data filtering

Gokhan Memik, William H. Mangione-Smith

October 2002 **Proceedings of the 2002 international conference on Compilers, architecture, and synthesis for embedded systems**

Full text available: pdf(234.17 KB) Additional Information: full citation, abstract, references, index terms

We propose and evaluate a data filtering method to reduce the power consumption of high-end processors with multiple execution cores. Although the proposed method can be applied to a wide variety of multi-processor systems including MPPs, SMPs and any type of single-chip multiprocessor, we concentrate on Network Processors. The proposed method uses an execution unit called Data Filtering Engine that processes data with low temporal locality before it is placed on the system bus. The execution co ...

Keywords: chip multiprocessors, data locality, network processors, power reduction, remote procedure call

20 On-chip communication architecture for OC-768 network processors

Faraydon Karim, Anh Nguyen, Sujit Dey, Ramesh Rao

June 2001 **Proceedings of the 38th conference on Design automation**

Full text available: pdf(244.89 KB) Additional Information: full citation, abstract, references, citings, index terms

The need for network processors capable of forwarding IP packets at OC-192 and higher data rates has been well established. At the same time, there is a growing need for complex tasks, like packet classification and differentiated services, to be performed by network processors. At OC-768 data rate, a network processor has 9 nanoseconds to process a minimum-size IP packet. Such ultra high-speed processing, involving complex memory-intensive tasks, can only be achieved by multi-CPU distribut ...

Terms used **network processor virtual sequentiality** Found **156** of **148,786**

Sort results by relevance Save results to a Binder Try an Advanced Search

Display results expanded form Search Tips Try this search in The ACM Guide

Open results in a new window

Results 1 - 20 of 156 Result page: **1** 2 3 4 5 6 7 8 next

Relevance scale

1 Balancing register allocation across threads for a multithreaded network processor

Xiaotong Zhuang, Santosh Pande

June 2004 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 2004 conference on Programming language design and implementation**, Volume 39 Issue 6

Full text available: pdf(429.85 KB) Additional Information: full citation, abstract, references, index terms

Modern network processors employ multi-threading to allow concurrency amongst multiple packet processing tasks. We studied the properties of applications running on the network processors and observed that their imbalanced register requirements across different threads at different program points could lead to poor performance. Many times application needs demand some threads to be more performance critical than others and thus by controlling the register allocation across threads one could impa ...

Keywords: multithreaded processor, network processor, register allocation

2 A fast string-matching algorithm for network processor-based intrusion detection system

Rong-Tai Liu, Nen-Fu Huang, Chih-Hao Chen, Chia-Nan Kao

August 2004 **ACM Transactions on Embedded Computing Systems (TECS)**, Volume 3 Issue 3

Full text available: pdf(571.00 KB) Additional Information: full citation, abstract, references, index terms

Network intrusion detection systems (NIDSs) are one of the latest developments in security. The matching of packet strings against collected signatures dominates signature-based NIDS performance. Network processors are also one of the fastest growing segments of the semiconductor market, because they are designed to provide scalable and flexible solutions that can accommodate change quickly and economically. This work presents a fast string-matching algorithm (called FNP) over the network proces ...

Keywords: Intrusion detection, network, pattern matching, processor

3 Session 1B: System-level exploration and design: NetBench: a benchmarking suite for network processors

Gokhan Memik, William H. Mangione-Smith, Wendong Hu

November 2001 **Proceedings of the 2001 IEEE/ACM international conference on Computer-aided design**

Full text available: pdf(77.08 KB) Additional Information: full citation, abstract, references, citings, index terms

In this study we introduce NetBench, a benchmarking suite for network processors. NetBench contains a total of 9 applications that are representative of commercial applications for network processors. These applications are from all levels of packet processing; Small, low-level code fragments as well as large application level programs are included in the suite.Using SimpleScalar simulator we study the NetBench programs in detail and characterize the network processor workloads. We also compare ...

4 Emerging areas: Programming challenges in network processor deployment

Chidamber Kulkarni, Matthias Gries, Christian Sauer, Kurt Keutzer

October 2003

Proceedings of the 2003 international conference on Compilers, architectures and synthesis for embedded systems

Full text available: pdf(234.71 KB) Additional Information: full citation, abstract, references, index terms

Programming multi-processor ASIPs, such as network processors, remains an art due to the wide variety of architectures and due to little support for exploring different implementation alternatives. We present a study that implements an IP forwarding router application on two different network processors to better understand the main challenges in programming such multi-processor ASIPs. The goal of this study is to identify the elements central to a successful deployment of such systems based on ...

Keywords: IPv4 forwarding, mapping, multi-threading, programming heterogeneous architectures, programming model, resource sharing

5 Session 7: embedded system techniques (2): Handling of packet dependencies: a critical issue for highly parallel network processors

Stephen Melvin, Yale Patt

October 2002 **Proceedings of the 2002 international conference on Compilers, architecture, and synthesis for embedded systems**

Full text available: pdf(221.66 KB) Additional Information: full citation, abstract, references, citings, index terms

Network processors are being asked to perform increasingly complex operations on packets of information at faster and faster rates. Because processor performance and memory cycle times are not keeping up with this demand, there is a fundamental need for simultaneous processing of multiple packets, and the degree of this parallelism is increasing. Sometimes a dependency exists between two packets currently being operated on, and as the ratio of packet processing time to packet transmission time i ...

Keywords: memory synchronization, multithreaded processors, network processors, packet dependencies, packet processing, parallel processing, processor architecture, thread level speculation

6 A pipelined memory architecture for high throughput network processors

Timothy Sherwood, George Varghese, Brad Calder

May 2003 **ACM SIGARCH Computer Architecture News , Proceedings of the 30th annual international symposium on Computer architecture**, Volume 31 Issue 2

Full text available: pdf(213.66 KB) Additional Information: full citation, abstract, references, citings

Designing ASICs for each new generation of backbone routers is a time intensive and fiscally draining process. In this paper we focus on the design of a programmable architecture for backbone routers, based on the manipulation of wide irregular memory words, that can provide a feasible design alternative to custom ASICs. We propose a pipelined memory design that emphasizes worst-case throughput over latency, and co-explore architectural tradeoffs with the design of several important network algo ...

7 Network processors: a perspective on market requirements, processor architectures and embedded S/W tools

P. Paulin, F. Karim, P. Bromley

March 2001 **Proceedings of the conference on Design, automation and test in Europe**

Full text available: pdf(269.19 KB) Additional Information: full citation, references, citings, index terms

8 Efficient use of memory bandwidth to improve network processor throughput

Jahangir Hasan, Satish Chandra, T. N. Vijaykumar

May 2003 **ACM SIGARCH Computer Architecture News , Proceedings of the 30th annual international symposium on Computer architecture**, Volume 31 Issue 2

Full text available: pdf(184.83 KB) Additional Information: full citation, abstract, references

We consider the efficiency of packet buffers used in packet switches built using network processors (NPs). Packet buffers are typically implemented using DRAM, which provides plentiful buffering at a reasonable cost. The problem we address is that a typical NP workload may be unable to utilize the peak DRAM bandwidth. Since the bandwidth of the packet buffer is often the bottleneck in the performance of a shared-memory packet switch, inefficient use of available DRAM bandwidth further reduces th ...

9 Memory optimization: Automatic data partitioning for the agere payload plus network processor
Steve Carr, Philip Sweany
September 2004 **Proceedings of the 2004 international conference on Compilers, architecture, and synthesis for embedded systems**
Full text available: pdf(195.89 KB) Additional Information: full citation, abstract, references, index terms

With the ever-increasing pervasiveness of the Internet and its stringent performance requirements, network system designers have begun utilizing specialized chips to increase the performance of network functions. To increase performance, many more advanced functions, such as traffic shaping and policing, are being implemented at the network interface layer to reduce delays that occur when these functions are handled by a general-purpose CPU. While some designs use ASICs to handle network functio ...

Keywords: network processors, partitioning, scheduling

10 Session S4.1: power in memory and network processors: Increasing power efficiency of multi-core network processors through data filtering
Gokhan Memik, William H. Mangione-Smith
October 2002 **Proceedings of the 2002 international conference on Compilers, architecture, and synthesis for embedded systems**
Full text available: pdf(234.17 KB) Additional Information: full citation, abstract, references, index terms

We propose and evaluate a data filtering method to reduce the power consumption of high-end processors with multiple execution cores. Although the proposed method can be applied to a wide variety of multi-processor systems including MPPs, SMPs and any type of single-chip multiprocessor, we concentrate on Network Processors. The proposed method uses an execution unit called Data Filtering Engine that processes data with low temporal locality before it is placed on the system bus. The execution co ...

Keywords: chip multiprocessors, data locality, network processors, power reduction, remote procedure call

11 On-chip communication architecture for OC-768 network processors
Faraydon Karim, Anh Nguyen, Sujit Dey, Ramesh Rao
June 2001 **Proceedings of the 38th conference on Design automation**
Full text available: pdf(244.89 KB) Additional Information: full citation, abstract, references, citings, index terms

The need for network processors capable of forwarding IP packets at OC-192 and higher data rates has been well established. At the same time, there is a growing need for complex tasks, like packet classification and differentiated services, to be performed by network processors. At OC-768 data rate, a network processor has 9 nanoseconds to process a minimum-size IP packet. Such ultra high-speed processing, involving complex memory-intensive tasks, can only be achieved by multi-CPU distribut ...

12 Session S6.2: compilers and program analysis: Experience with a retargetable compiler for a commercial network processor
Jinhwan Kim, Sungjoon Jung, Yunheung Paek, Gang-Ryung Uh
October 2002 **Proceedings of the 2002 international conference on Compilers, architecture, and synthesis for embedded systems**
Full text available: pdf(275.87 KB) Additional Information: full citation, abstract, references, citings, index terms

The Paion PPII network processor is designed to meet the growing need for new high bandwidth network equipment. In order to rapidly reconfigure the processor for frequently varying internet services and technologies, a high performance compiler is urgently needed. Albeit various code generation techniques have been proposed for DSPs or ASIPs, we experienced these techniques are not easily tailored towards the target Paion PPII processor due to striking architectural differences. First, we will s ...

Keywords: compiler, network processor, non-orthogonal architecture

13 Improving route lookup performance using network processor cache
Kartik Gopalan, Tzi-cker Chiueh
November 2002 **Proceedings of the 2002 ACM/IEEE conference on Supercomputing**
Full text available: pdf(282.33 KB) Additional Information: full citation, abstract, references, index terms

Earlier research has shown that the route lookup performance of a network processor can be significantly improved by caching ranges of lookup/classification keys rather than individual keys. While the previous work focused specifically on reducing capacity misses, we address two other important aspects - (a) reducing conflict misses and (b) cache consistency during frequent route updates. We propose two techniques to minimize conflict misses that aim to balance the number of cacheable entries ma ...

14 A network project course based on network processors

Peter Steenkiste

January 2003 **ACM SIGCSE Bulletin , Proceedings of the 34th SIGCSE technical symposium on Computer science education**, Volume 35 Issue 1

Full text available: pdf(160.53 KB) Additional Information: full citation, abstract, references, index terms

A difficult problem in networking courses is to find hands-on projects that have the right balance between the level of realism and complexity. This is especially true for projects that focus on the internal functionality of routers and other network devices. We developed a capstone course called "Network Design and Evaluation" that uses a network processor-based platform for networking projects. This platform is more realistic than traditional approaches based on software emulation environments ...

Keywords: network internals, network project, pedagogy

15 Building a robust software-based router using network processors

Tammo Spalink, Scott Karlin, Larry Peterson, Yitzchak Gottlieb

October 2001 **ACM SIGOPS Operating Systems Review , Proceedings of the eighteenth ACM symposium on Operating systems principles**, Volume 35 Issue 5

Full text available: pdf(1.49 MB) Additional Information: full citation, abstract, references, citings, index terms

Recent efforts to add new services to the Internet have increased interest in software-based routers that are easy to extend and evolve. This paper describes our experiences using emerging network processors---in particular, the Intel IXP1200---to implement a router. We show it is possible to combine an IXP1200 development board and a PC to build an inexpensive router that forwards minimum-sized packets at a rate of 3.47Mpps. This is nearly an order of magnitude faster than existing pure PC-base ...

16 Student posters from SIGCOMM 2001: Network processors: flexibility and performance for next-generation networks

Tilman Wolf

January 2002 **ACM SIGCOMM Computer Communication Review**, Volume 32 Issue 1

Full text available: pdf(93.45 KB) Additional Information: full citation, references, index terms

17 Network processors: a progress report

Nevin Heintze

September 2004 **Proceedings of the fourth ACM international conference on Embedded software**

Full text available: pdf(122.92 KB) Additional Information: full citation, abstract

Network processors generated huge excitement, interest and investment in the late 90's and into 2000, promising to replace fixed-function ASIC chips in networking equipment by high-performance programmable devices. The sheer performance requirements of these devices led to a flurry of activity in both processor architecture, software and especially the interaction between architecture and software, with many diverse approaches competing for survival. The subsequent collapse of the market for com ...

18 Taming the IXP network processor

Lal George, Matthias Blume

May 2003 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 2003 conference on Programming language design and implementation**, Volume 38 Issue 5

Full text available: pdf(159.27 KB) Additional Information: full citation, abstract, references, citings, index terms, review

We compile Nova, a new language designed for writing network processing applications, using a back end based on integer-linear programming (ILP) for register allocation, optimal bank assignment, and spills. The compiler's optimizer employs CPS as its intermediate representation; some of the invariants that this IR guarantees are essential for the formulation of a practical ILP model.Appel and George used a similar ILP-

based technique for the IA32 to decide which variables reside in registers but ...

Keywords: Intel IXA, bank assignment, code generation, integer linear programming, network processors, programming languages, register allocation

19 Special Session on Network Processors: An Industrial Perspective: Network processing in content inspection applications

Feliks J. Welfeld

September 2001 **Proceedings of the 14th international symposium on Systems synthesis**

Full text available: pdf(175.50 KB) Additional Information: full citation, abstract, citings, index terms

This paper will show how NP can be used for complicated tasks including advanced classification of packets. Advanced classification requires the NP to look beyond the layer 3-4 header of the protocol, deep into the session and application layer payload. Dubbed "content inspection", such advanced classification requires specialized high-performance ICs, and a powerful and sophisticated set of tools for programming the ICs. These tools enable users of advanced classifiers to describe protocols usi ...

20 Special Session on Network Processors: An Industrial Perspective: Embedded systems technologies for application-specific architecture platforms

Pierre G. Paulin

September 2001 **Proceedings of the 14th international symposium on Systems synthesis**

Full text available: pdf(97.70 KB) Additional Information: full citation, abstract, references, index terms

This talk addresses the emerging architectural platform approach to improve time-to-market, and the design automation technologies needed to support it. Architectural platforms can be defined as a domain- or application-specific base design which is easily configured to the specific needs of a given market. An experimental network processing platform developed at STMicroelectronics [1] will be presented to illustrate the concept. Key platform automation challenges will be introduced.In order to ...